Amendments to the Specification:

Please replace the paragraph beginning at line 21 of page 10 with the following amended paragraph:

P 1

Fig. 2 shows a system 15 which is used with the present invention. It shows a plurality of host computers or processors 12, [[in]] on which a client process 11A would reside [[on]], connected to data storage system 14. Host processors 12, which are typically digital processing units which may include one or more central processing units (CPUs) in main memory. As indicated they might be, for example, UNIX workstations, or NT workstations.

Please replace the paragraph beginning at line 5 of page 11 with the following amended paragraph:

P 2

In general, data storage system 14 contains a shared memory 22 that is accessible to all of the host processors 12 that are connected to the data storage system 14. Host processors 12 are each connected to data storage system 14 through respective host connection 16', which, as stated earlier, is typically a high-speed data communication link such as SCSI. To simplify the discussion, only single host connection is shown for each host processor 12. It should be understood, however, that there could in fact be multiple connections between the data storage system 14 and a host processor 12. Data storage system 14 contains the physical memory in which data is stored. The particular manner in which the physical memory within the data storage system is implemented and how it is partitioned is not of central importance to the present invention. An example of a commercially available product to implement the data storage system 14 is the Symmetrix family of products from EMC Corporation, which are high-performance integrated cache disk arrays (ICDA) designed for on-line data storage. The following details about the internal structure and operation of the data storage system 14 generally apply to the Symmetrix data storage systems. However, it should be understood that

other designs known to persons skilled in the art may also be used to implement data storage system 14.

Please replace the paragraph beginning at line 4 of page 14 with the following amended paragraph:

Computer 12 [[which]] contains the server process 11 of the backup and restore application. The server process 11 includes other software modules. ~~Included~~ For example, included with the server process are three other software modules. In order to allow the server process 11 to facilitate a communication over the network 18 to the client process 11A, a communication mechanism 58 must be used. As in earlier network based backup and restore software applications, the preferred communication mechanism is sockets and, particularly TCP/IP sockets used to create and establish a communication to the client process 11A. Also in the server process 11 is a second communications mechanism shown at 60. In the preferred embodiment of the invention, sockets are used as a communication mechanism to facilitate communications from the server or client processes 11 and 11A through the data storage system 14 to the client or server processes 11A and 11. In the preferred embodiment of the invention, the sockets 60 are STP (Symmetrix Transport Protocol) or SSLsockets, as further described in EMC Corporation's, assignee of the present application, pending patent application entitled "Communication Mechanism and Method for Easily Transferring Information Between Processes" filed on September 29, 1997, having serial number 08/939,772. The manner in which communications and data are transferred through the data storage system 14 with the use of communication mechanism 60 is further described in EMC Corporation's pending patent application entitled "Method and Apparatus for Transfers Employed in a Data Storage System" filed on December 30, 1997, [1997] and having serial number 09/000,540. Lastly, included in the server process 11 is a layer of software termed STP or SSLConnect, in the preferred embodiment of the invention, and shown at 54. In the preferred embodiment of the invention, this software allows the server process 11 to determine the mode or type of a particular socket call. For example, if the socket call is a normal or typical TCP/IP socket call it will process that socket call as a "normal" socket

request, and use the created socket communication mechanism 58 to create and/or

facilitate communications over the network 18. If however, the socket call is one

specially designed for use with the data storage system 14, the STP or SSLConnect

software will determine that the particular type or mode of that socket call is a socket call

specially designed for data storage system 14, and will process that request as a socket

call specially utilized for data storage system 14. Thus, the server process 11 will use the

socket communication mechanism 60 to create and/or facilitate communications through

the data storage system 14.

Please replace the paragraph beginning at line 3 of page 20 with the following amended

paragraph:

It should be noted that, once the client process sends the file descriptor to the

server process at step 122, the client process falls into a loop. The client process is then

ready to accept another connection. Once the server process receives the file descriptor,

it opens another socket on the same WKP. The numbers contained with the socket are

simply to indicate different sockets. The socket is opened at step 94. As previously

shown, the socket is followed by bind and connect commands at steps 96 and 98

respectively. The client accepts the connect at step 98 at step 124. It should be noted that

all of this is done on the WKP. Once the client process has accepted the connection

request at step 124, the server process sends to the client process, at step 100, the file

descriptor, which is received by the client process at step 126. Step 100 allows the server

to send the file descriptor to the client to let the client know it is in connection with the

same server process. The second of the two connections the client process has been told

by the configuration file that it going to receive is now completed. Because two

connections have been made, the server process and the client process in essence have

two separate channels for use in communication with each other. Typically, one of the

communication channels is used for data communications, while the second is used for

error communication.

Stockenberg *et al.*
09/052,325

Please replace the paragraph beginning at line 3 of page 20 with the following amended paragraph:

When the STPsocket call is issued at step 162, it is going to be bound to a dynamically allocated port. At step 162 the socket is bound to the dynamically allocated port. A STPlisten command is issued at step 166. At step 168, the STPgetsockname command is issued, the information containing the value of the dynamically allocated port is sent back to the server process at step 170. The server process receives [[the]] this information at step 140 and then establishes a socket beginning at step 142 to begin to establish a connection through the data storage system. Once the socket is created at step 142 with the STPsocket command, it is once again bound, to the dynamically allocated port.

Please replace the paragraph beginning at line 8 of page 24 with the following amended paragraph:

Since the server process has the information about the dynamically allocated port[[.]], [[The]] the server process is ready to open up a socket for that dynamically allocated port. The socket is opened at step 228. The server process indicates its readiness for connection on the dynamically allocated port at step 234. The client process indicates its acceptance of the socket created at step 228 at step 198 through an accept command. As the server process becomes aware that the client process has accepted the socket on a dynamically allocated port, it sends, at step 236, a message identifying the STG group to be used. This information is received by the server process at step 200. It should be understood that the socket created on the WKP is really not being used for anything, its purpose is really just to initiate the steps necessary to establish the connection on the dynamically allocated port, which as will be seen, is really used to establish the connection through the data storage system.